

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A method of managing queue entries, comprising:
  - storing addresses in a first queue entry as a circular linked list wherein the first entry points to a subsequent entry and to a final entry, each of the stored addresses referring to a stored data buffer and including a cell count that indicates a number of cells contained in the data buffer;
  - retrieving a first address from the first queue entry; and
  - modifying the linked list of addresses of the first queue entry based on the cell count of the first address retrieved, including decrementing the cell count of the first address each time the first address is retrieved.
2. (Canceled)
3. (Currently amended) The method of claim 1 ~~2~~, further comprising:
  - determining the cell count is zero; ~~and~~
  - ~~setting a second address as the first address of the first queue entry.~~
4. (Original) The method of claim 3, wherein storing addresses further comprises:
  - setting the first address as the head address of the first queue entry; and
  - linking a second address to the first address of the first queue entry.
5. (Original) The method of claim 4, wherein linking the second address to the first address further comprises:

setting the second address as a tail address of the first queue entry.

6. (Original) The method of claim 5, further comprising:

linking a third address to the first queue entry by storing the third address in the location indicated by the tail address.

7. (Original) The method of claim 5, further comprising:

incrementing a queue count each time an address is linked to the first queue entry

8. (Original) The method of claim 4, wherein the first queue entry is stored as part of a queue array having a plurality of linked queue entries.

9. (Currently amended) An article comprising a machine-readable medium that stores machine-executable instructions for managing a queue array, the instructions causing a machine to:

store addresses in a first queue entry as a circular linked list wherein the first entry points to a subsequent entry and to a final entry, each of the stored addresses referring to a stored data buffer and including a cell count that indicates a number of cells contained in the data buffer; retrieve a first address from the first queue entry; and

modify the linked list of addresses of the first queue entry based on the cell count of the first address retrieved, including decrementing the cell count of the first address each time the first address is retrieved.

10. (Canceled)

11. (Currently amended) The article of claim 9 ~~10~~, further comprising instructions causing a machine to:

determine the cell count is zero ~~;~~ ~~and~~

~~set a second address as the first address of the first queue entry.~~

12. (Original) The article of claim 11, wherein storing further comprises instructions causing a machine to:

set the first address as the head address of the first queue entry; and  
link a second address to the first address of the first queue entry.

13. (Original) The article of claim 12, wherein linking comprises setting the second address as a tail address of the first queue entry.

14. (Original) The article of claim 13, further comprising instructions causing a machine to:  
link a third address to the first queue entry by storing the third address in the location indicated by the tail address.

15. (Original) The article of claim 13, further comprising instructions causing a machine to:  
increment a queue count each time an address is linked to the first queue entry

16. (Original) The article of claim 12, wherein the first queue entry is stored as part of a queue array having a plurality of linked queue entries.

17. (Currently amended) An apparatus, comprising:

a first storage device for holding queue entries;  
a second storage device for holding data packets;  
a memory that stores executable instructions; and  
a processor that executes the instructions to:

store addresses in a first queue entry as a circular linked list wherein the first entry points to a subsequent entry and to a final entry, the addresses referring to a stored data buffer and including a cell count that indicates a number of cells contained in the data buffer;

retrieve a first address from the first queue entry, and

modify the linked list of addresses of the first queue entry based on the cell count of the first address retrieved, including decrementing the cell count of the first address each time the first address is retrieved.

Claim 18. (Canceled)

19. (Currently amended) The apparatus of claim 17 ~~18~~, wherein instructions to modify comprise instructions to:

determine the cell count is zero; ~~and~~  
~~set a second address as the first address of the first queue entry.~~

20. (Original) The apparatus of claim 19, wherein instructions to store addresses comprises instructions to:

set the first address as the head address of the first queue entry; and  
link a second address to the first address of the first queue entry.

21. (Original) The apparatus of claim 20, wherein instructions to link comprises instructions to:  
set the second address as a tail address of the first queue entry.

22. (Original) The apparatus of claim 21, further comprising instructions to:  
link a third address to the first queue entry by storing the third address in the location indicated by the tail address.

23. (Original) The apparatus of claim 21, further comprising instructions to:  
increment a queue count each time an address is linked to the first queue entry.

24. (Original) The apparatus of claim 20, further comprising:  
a storage medium, the first queue entry being stored on the storage medium as part of a queue array having a plurality of linked queue entries.

25. (Currently amended) A processing system for managing queue

entries comprising:

a processor;

a memory to store queue entries; and

a storage-medium accessible by the processor to store executable instructions, which when accessed by the processor causes the processor to:

store addresses in a first queue entry as a circular linked list wherein the first entry points to a subsequent entry and to a final entry, each of the stored addresses referring to a stored data buffer and including a cell count that indicates a number of cells contained in the data buffer;

retrieve a first address from the first queue entry; and

modify the linked list of addresses of the first queue entry based on the cell count of the first address retrieved, including decrementing the cell count of the first address each time the first address is retrieved.

Claim 26. (Canceled)

27. (Currently amended) The method of claim 25 26, further comprising instructions, which when accessed by the processor causes the processor to:

determine the cell count is zero; ~~and~~

~~set a second address as the first address of the first queue entry.~~

28. (Previously presented) The system of claim 27, wherein storing addresses further comprises:

setting the first address as the head address of the first queue entry; and

linking a second address to the first address of the first queue entry.

29. (Previously presented) The system of claim 28, wherein linking the second address to the first address further comprises instructions, which when accessed by the processor causes the processor to:

setting the second address as a tail address of the first queue entry.

30. (Previously presented) The system of claim 29, further comprises instructions, which when accessed by the processor causes the processor to:

link a third address to the first queue entry by storing the third address in the location indicated by the tail address.

31. (Previously presented) The system of claim 29, further comprises instructions, which when accessed by the processor causes the processor to:

increment a queue count each time an address is linked to the first queue entry.